Computer graphics algorithms and techniques for Monte Carlo Simulation

Stewart T. Boogert
University of Manchester

Laurie Nevay
CERN

Aims

- Enable complex geometric description and simulation with **even less** people resources than very modest HEP experiments
- Improve transfer of geometry between commonly used codes and CAD
- Promote maintainability and reuse of geometry
- Reduce time between specifying task and physics results

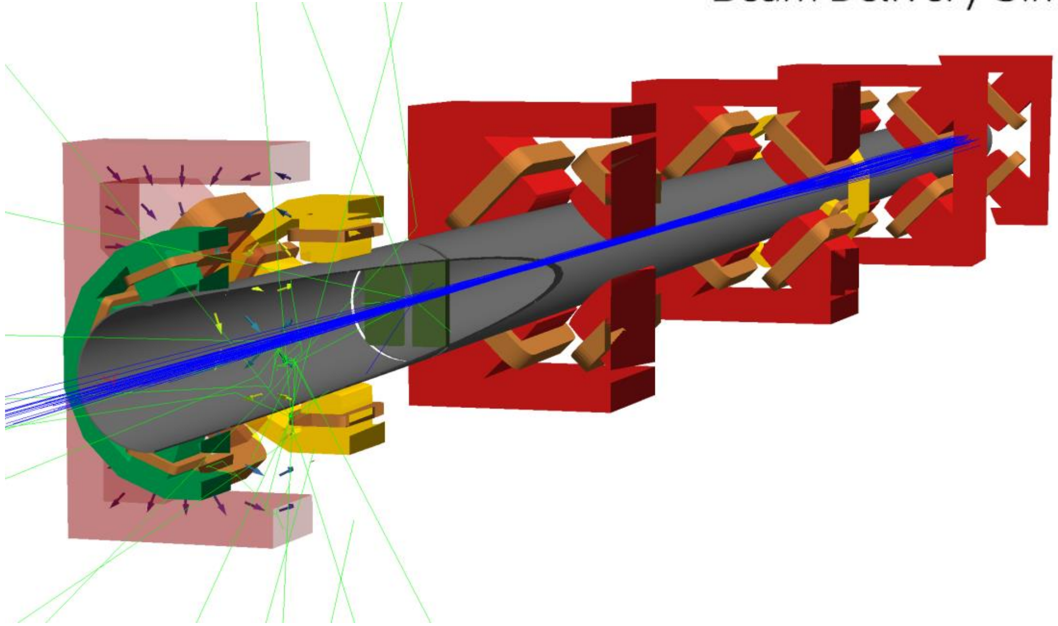
Journey up to this point

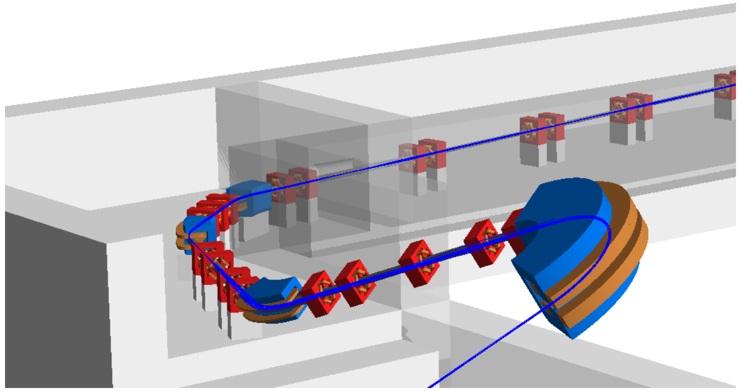
Involvement in Geant4 started with trying to understand accelerator backgrounds, machine detector interface or dense beamlines

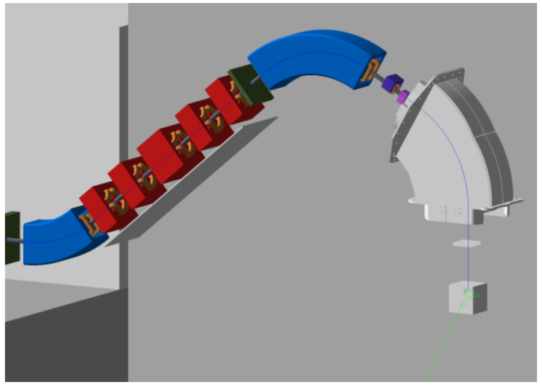
- Accelerator and Medical physics requires Monte Carlo simulation
  - Accelerators (Geant4, FLUKA, MCNP etc)
  - Medical (Geant4)
- Group developed multiple codes
  - Beam delivery simulation (BDSIM)
  - Pyg4ometry
  - VTK visualization in Geant4
  - CGAL Booleans for Geant4

Beam delivery simulation

- Geant4 based application to simulate beam lines
  - Fast accelerator tracking in beam pipe
  - Full physics outside
  - Proton therapy systems, CERN beamline, EU-XFEL, PSI beamlines, FPF, FASER, HIKE/Shadows
  - Design for ion radiobiology facility

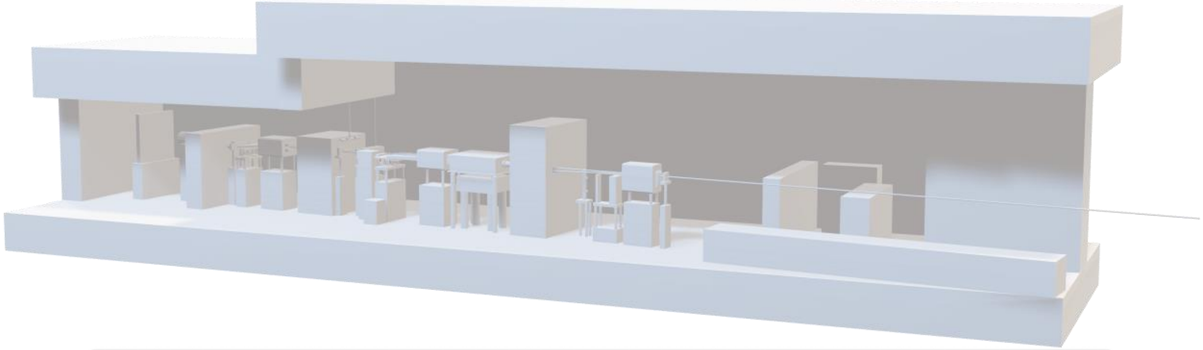




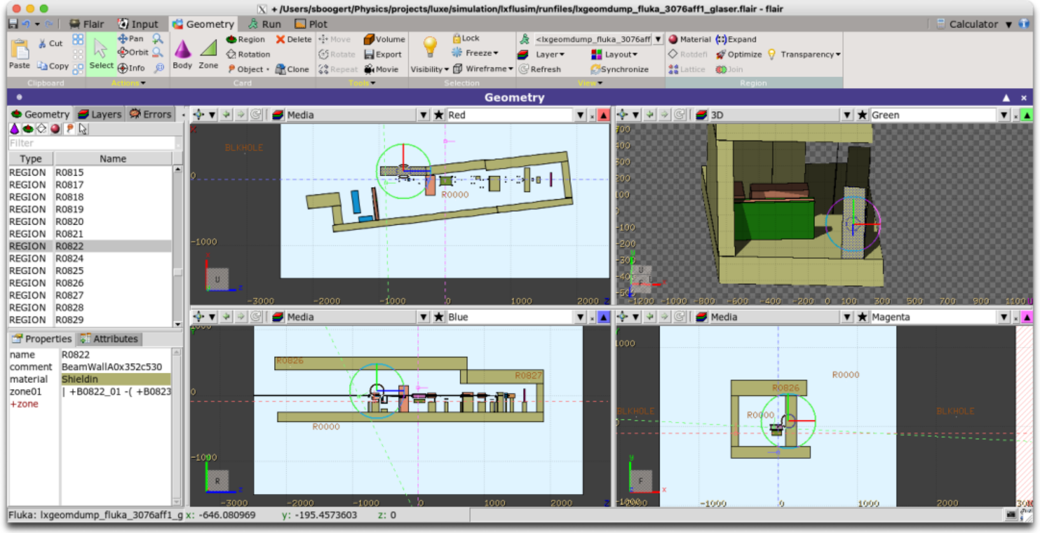


http://www.pp.rhul.ac.uk/bdsim/manual/

Pyg4ometry



- Python library to manipulate and convert geometry
- Basically
  - Pythonic API to G4 geometry
  - Ability to read/write gdml/inp/step
  - Geometry manipulation via CGAL
  - Visualisation in VTK
- Not a replacement for DD4Hep/GDML
- Expt users : Legend/Moller

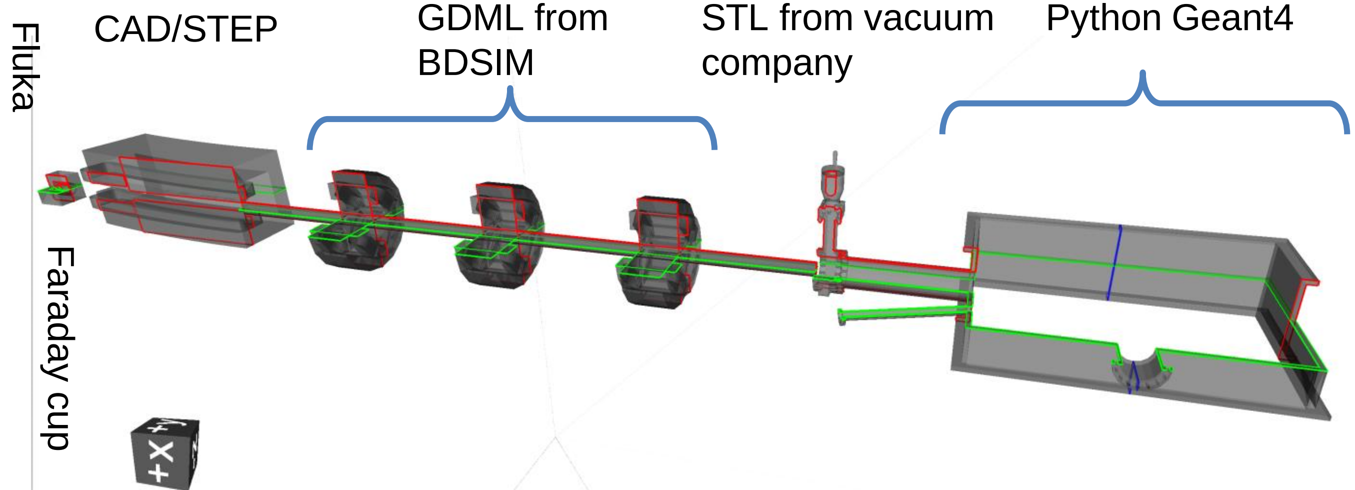

+ /Users/sboogert/Physics/projects/luxe/simulation/lxflusim/runfiles/lxgeomdump_fluka_3076aff1_glaser.flair - flair

Type	Name
REGION	R0815
REGION	R0817
REGION	R0818
REGION	R0819
REGION	R0820
REGION	R0821
REGION	R0822
REGION	R0824
REGION	R0825
REGION	R0826
REGION	R0827
REGION	R0828
REGION	R0829

Properties / Attributes

name	R0822
comment	BeamWallA0x352c530
material	Shieldin
zone01	| +B0822_01 -(+B0823
+zone	

Fluka: lxgeomdump_fluka_3076aff1_g x: -646.080969 y: -195.4573603 z: 0


Fluka

CAD/STEP

GDML from BDSIM

STL from vacuum company

Python Geant4

Faraday cup

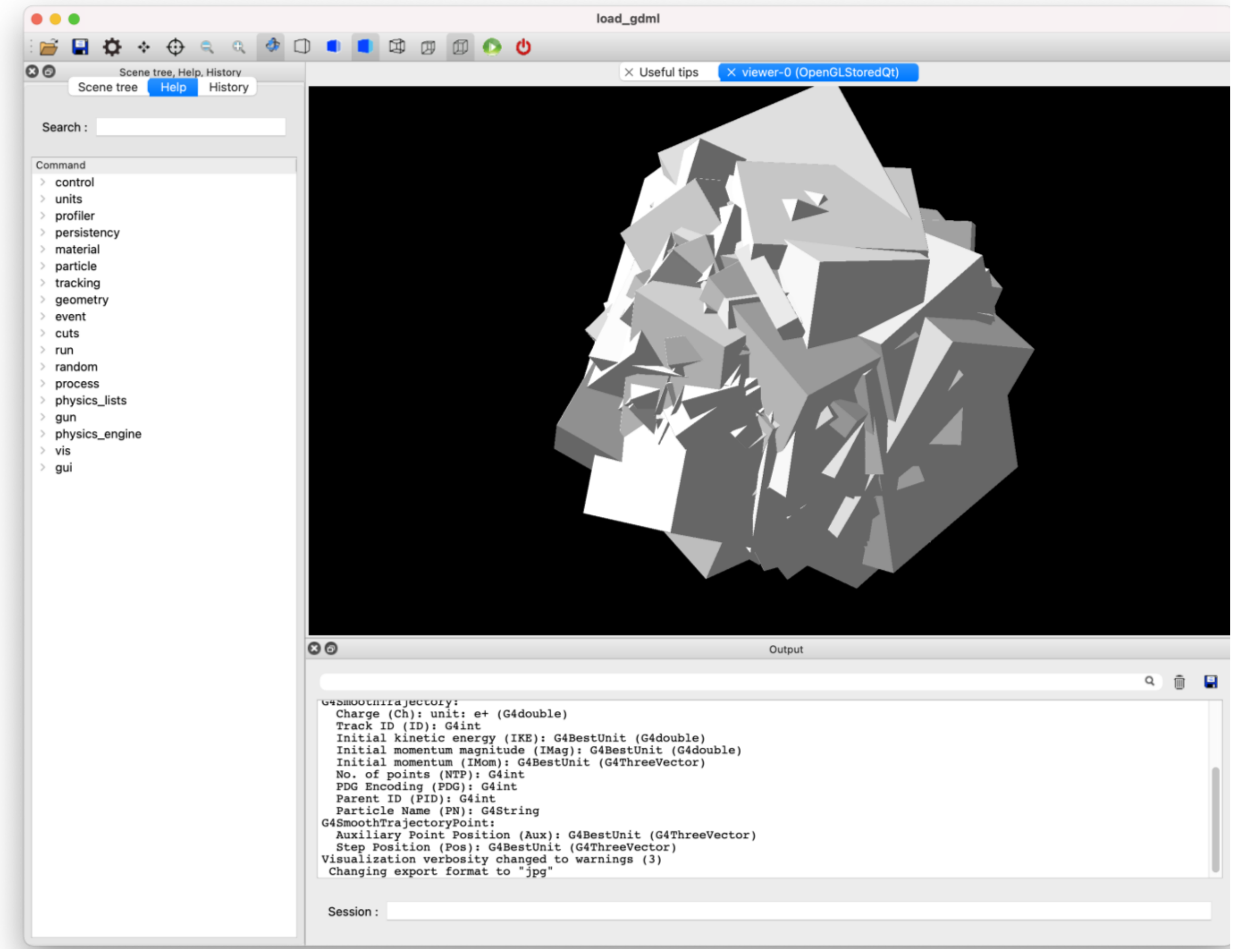
https://github.com/g4edge/pyg4ometry

MANCHESTER
1824
The University of Manchester

CGAL

CGAL Boolean processing in Geant4

- Geant4 default Boolean operations are not robust
- Recent release of Geant4 to allow replacement Boolean processor
- CGAL license does not allow distribution with Geant4

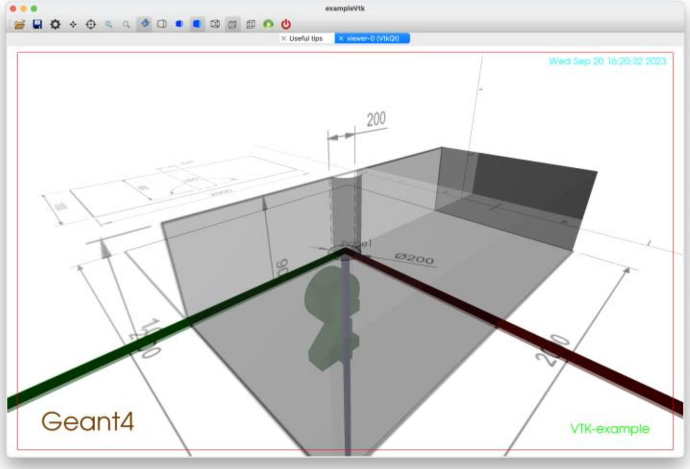


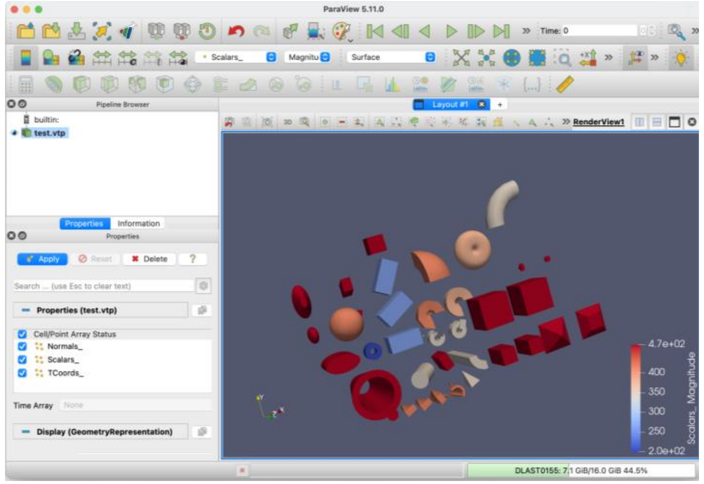
```
G4SmoothTrajectory:
  Charge (Ch): unit: e+ (G4double)
  Track ID (ID): G4int
  Initial kinetic energy (IKE): G4BestUnit (G4double)
  Initial momentum magnitude (IMag): G4BestUnit (G4double)
  Initial momentum (IMom): G4BestUnit (G4ThreeVector)
  No. of points (NTP): G4int
  PDG Encoding (PDG): G4int
  Parent ID (PID): G4int
  Particle Name (PN): G4String
G4SmoothTrajectoryPoint:
  Auxiliary Point Position (Aux): G4BestUnit (G4ThreeVector)
  Step Position (Pos): G4BestUnit (G4ThreeVector)
Visualization verbosity changed to warnings (3)
 Changing export format to "jpg"
```

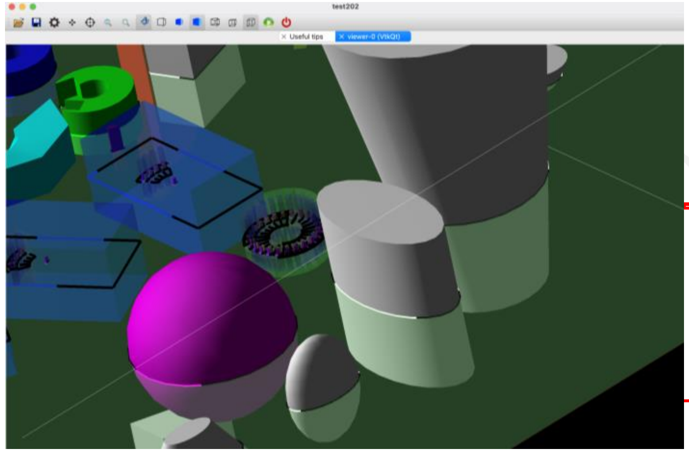
https://github.com/g4edge/g4cgal

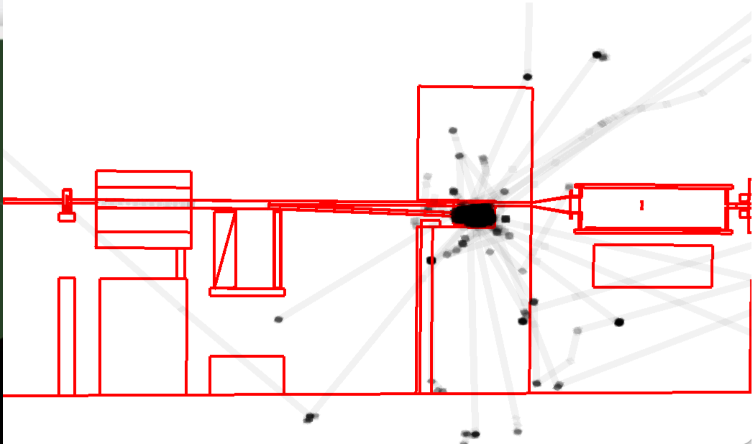
VTK visualisation driver for Geant4

- The largest accelerator models could not be visualised using standard tools of geant4
  - Developed pipelined VTK render for Geant4
  - Good performance
  - Well developed for next G4 release









Techniques in modern computer graphics

- Accelerator (CAD, FLUKA)
  - Half space decomposition of solids
  - Loading of FLUKA/MCNP/PHITS geometry into Geant4
  - Many non-Geant4 codes use a form of Constructive solid geometry (CSG) based on finite and infinite half spaces

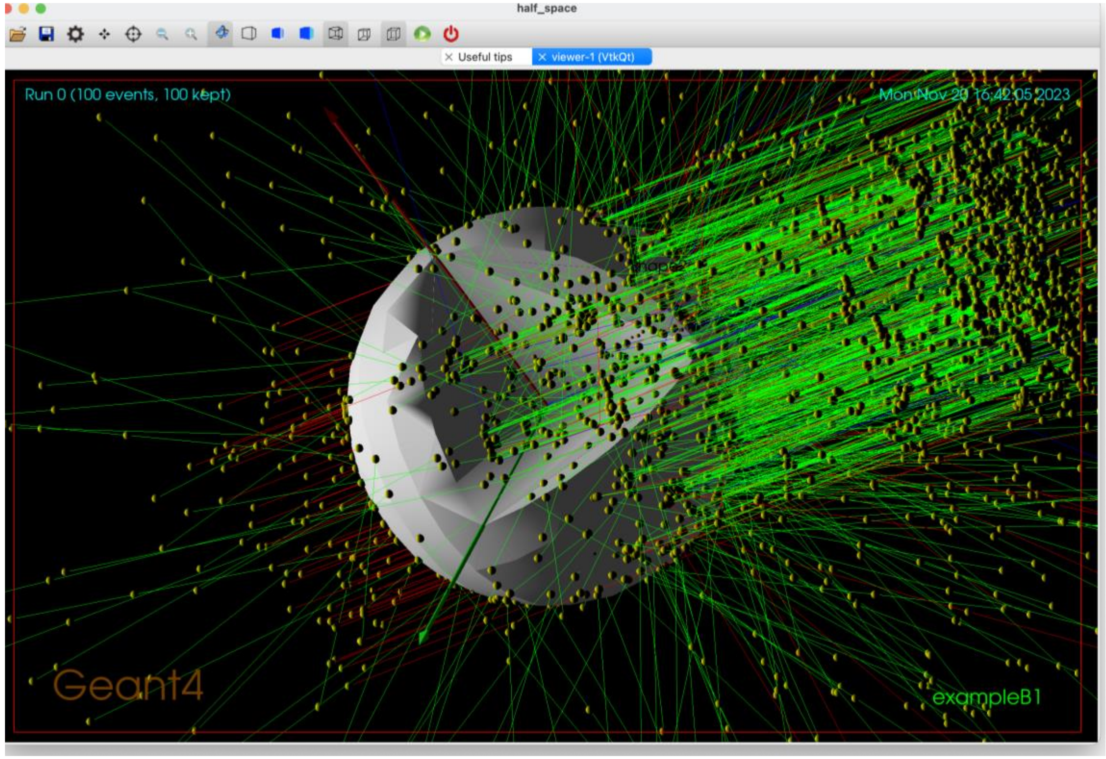
Need better half space geometry and tracking

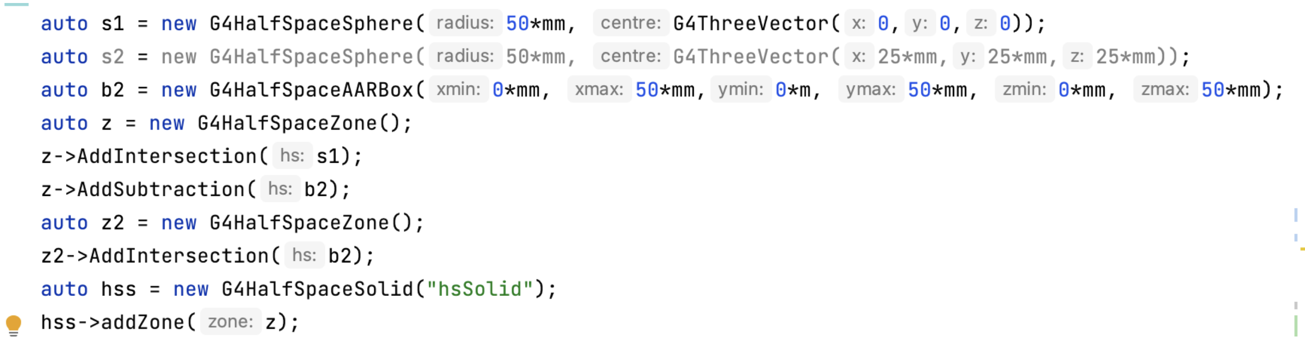
- Medical
  - Phantoms are complex geometries not well described by standard solids
  - Cubical voxel geometries generated by imaging systems also have limitations
    - How to deform geometry?
  - Reference phantoms are mesh based e.g ICRP145

Need better mesh geometry tracking

Prototype half space tracker being developed for Geant4

- Half spaces are typically quadrics (2d quadratic surfaces)
  - e.g. ellipsoid, hyperbolic cylinder etc.
  - No fundamental reason why higher orders cannot be included (apart from computational speed and accuracy)
- Relax requirement that geometry has to be in disjunctive normal form (DNF) i.e. union of convex solids
- **Signed distance fields (SDFs) perform all of the Boolean logic**
  - Helpful for distance to inside calculations
  - Difficult/annoying to construct for some quadrics (requires 3rd/4th order polynomial roots)
  - Use to determine if intersection is valid





```
auto s1 = new G4HalfSpaceSphere( radius: 50*mm,  centre: G4ThreeVector( x: 0, y: 0, z: 0));
auto s2 = new G4HalfSpaceSphere( radius: 50*mm,  centre: G4ThreeVector( x: 25*mm, y: 25*mm, z: 25*mm));
auto b2 = new G4HalfSpaceAARBox( xmin: 0*mm,  xmax: 50*mm, ymin: 0*m,  ymax: 50*mm,  zmin: 0*mm,  zmax: 50*mm);
auto z = new G4HalfSpaceZone();
z->AddIntersection( hs: s1);
z->AddSubtraction( hs: b2);
auto z2 = new G4HalfSpaceZone();
z2->AddIntersection( hs: b2);
auto hss = new G4HalfSpaceSolid("hsSolid");
hss->addZone( zone: z);
```

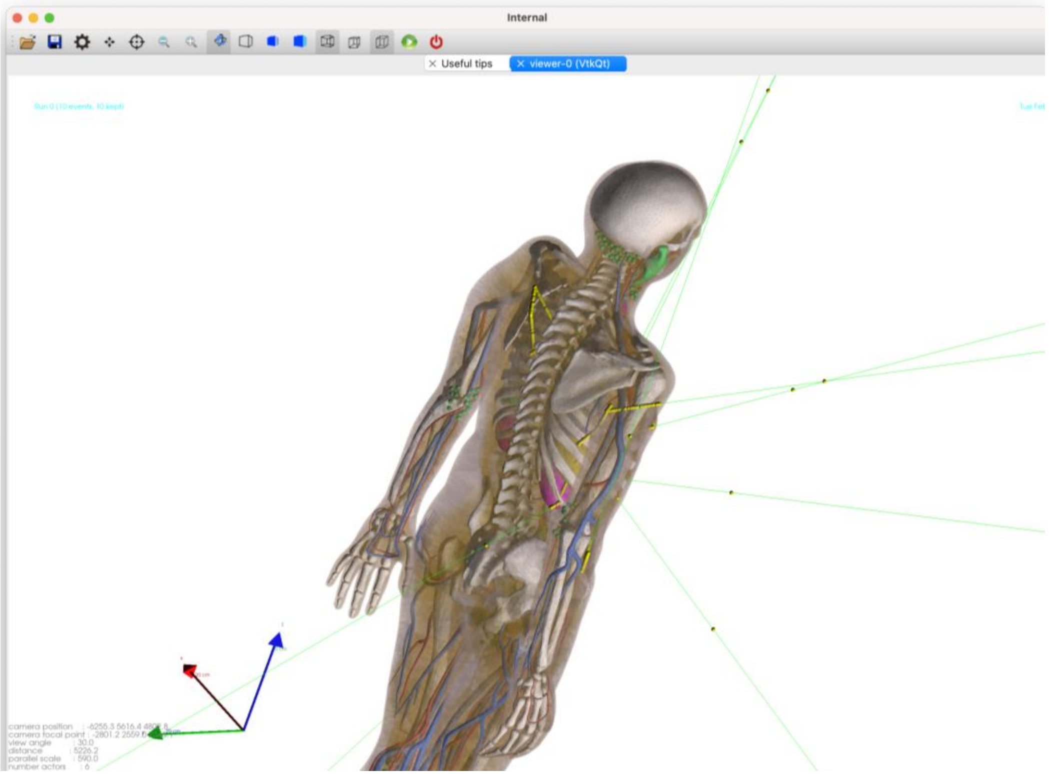
Half space tracker possibilities

- Bounded volume hierarchies to speed up
  - Compare performance with DNF H–rep and one with a richer Boolean structure
- Loaders (Antlr) for common formats to improve inter-operation
  - MCNP, FLUKA, PHITs…

- CAD conversion to H-rep is a *relatively* solved problem
  - Still an issue with NURBS or other cubical splines
  - Should there be a review of why we avoid these surfaces, i.e iterative solutions?

Medical phantoms

- ICRP145 in geant4 rendered using VTK
  - Typically, due to voxelization, users tetrahedralise the surface mesh
  - Large files, difficult to manipulate
  - Impossible to deform
- Applications in personalised health care

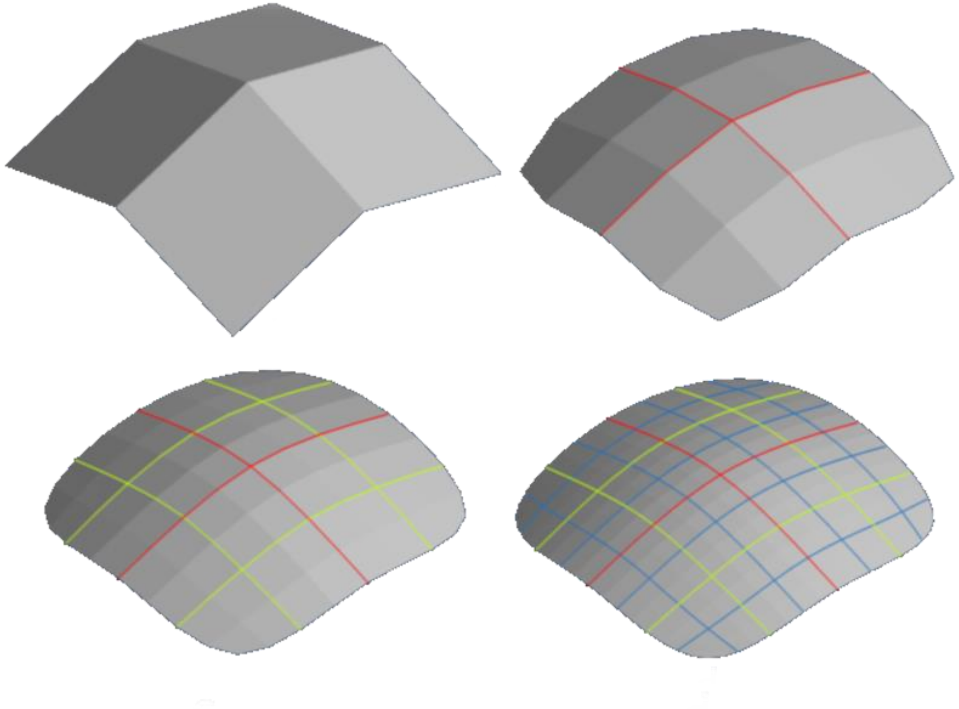


```
camera position    : -6255.3 5616.4 4808.5
camera focal point : -2801.2 2559.6
view angle         : 30.0
distance           : 5226.2
parallel scale     : 590.0
number actors      : 6
```

MANCHESTER 1824
The University of Manchester

Subdivision meshes

- Modelling in games/VFX typically do not use assets at full polygon count
  - Algorithmic subdivision of geometry and if fine detail is required geometry shading
  - Significantly easier to deform
  - Lower memory for phantom models
- Use of something like OpenSubdiv
- Quad dominated meshes
- Investigate subdivision during tracking and caching geometry



No Subusurf
Level 1
Level 2
Level 3

CGAL algorithms in Geant4

- Boolean processing in Geant4 is a good start
- Other algorithms
  - Tetrahedralisation
  - Mesh repair
  - Decomposition
  - Point cloud reconstruction
- Same 3D geometry linked between Geant4-multiphysics simulations

- Ultimately should direct use of triangular meshes be used to fully enable complex geometry
  - BVHs a la Embree
  - Or DagMC
  - Need to fully explore for G4TesselatedSolid

Other algorithms

- Differentiable rendering (e.g. Mitsuba3)
- Is there an equivalent of level of detail (LoD) for MC
  - Sources distant from a detector
- Switch geometry description during simulation?

- Wide range of variance reduction techniques in photon rendering

MANCHESTER 1824
The University of Manchester

Programme of work

- **Complete H-rep tracker**
  - Data loaders for PHITS/MCNP etc
  - Benchmark
- Develop CAD H-rep converter
  - Examples already existing from neutronics/nuclear community
- **Prototype subdivision tracker**
  - Embree based example started for tessellated solids
- **Direct mesh tracking**
  - Binary space partitioning trees
  - Bound volume hierarchies
- Direct SDF tracking
  - Allows a different form of modelling
  - Sphere tracing is not efficient for MC workflows
- Develop CGAL workflow within Geant4
  - Tetrahedralisation
  - Mesh repair
  - Mesh simplification (for simulation optimization)

https://github.com/g4edge/g4cgal

Summary (my opinion)

- Ok, ok so not quite HEP, not even HEP adjacent
- Rich programe of possible work around half space representations, subdivision meshes, direct mesh tracking, SDFs and differentiable MC
- Small accelerator group attacking these problems as need arises and with available capacity
- Need a rich set of tools in this area to better allow us as a simulation community to track computer graphics developments
  - Formats (GLTF, USD)
  - Simulation framework (e.g. Geant4) with CGAL, Embree, OpenSubdiv, OpenVDB, Mitsuba3 integrated